



FACULTÉ DES SCIENCES SAINT-ÉTIENNE

RAPPORT PROJET PROGRAMMATION IMPÉRATIVE

CHICHBEK

MEZIANE GHILAS

MENAA ABDERREZAK

L2 Informatique

1- Identité graphique :

Nous avons tout d'abord conçu un logo et une identité graphique pour notre jeu. Le logo a été fait sur Adobe Photoshop , ainsi que le fond du jeu. Puis une conception du menu a été aussi également designée par nos soins , afin de mieux pouvoir se projeter dans notre jeu. Avec ça , il a aussi fallu concevoir tous les boutons et éléments graphiques. ci dessous un aperçu de toute cette identité qui donne un certain charme à notre jeu :



FIGURE 1 – menu

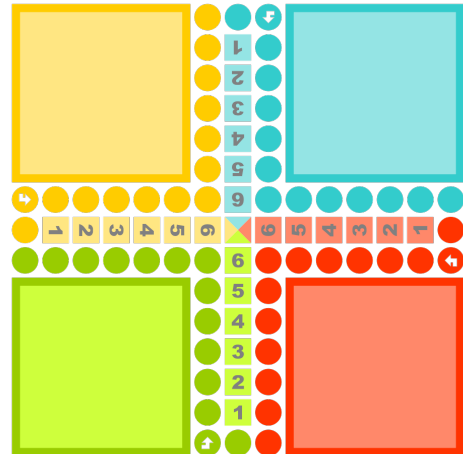


FIGURE 2 – plateau

lancer le dé

FIGURE 3 – bouton lancer dé

NOUVELLE PARTIE

FIGURE 5 – bouton nouvelle partie

MEILLEURS SCORES

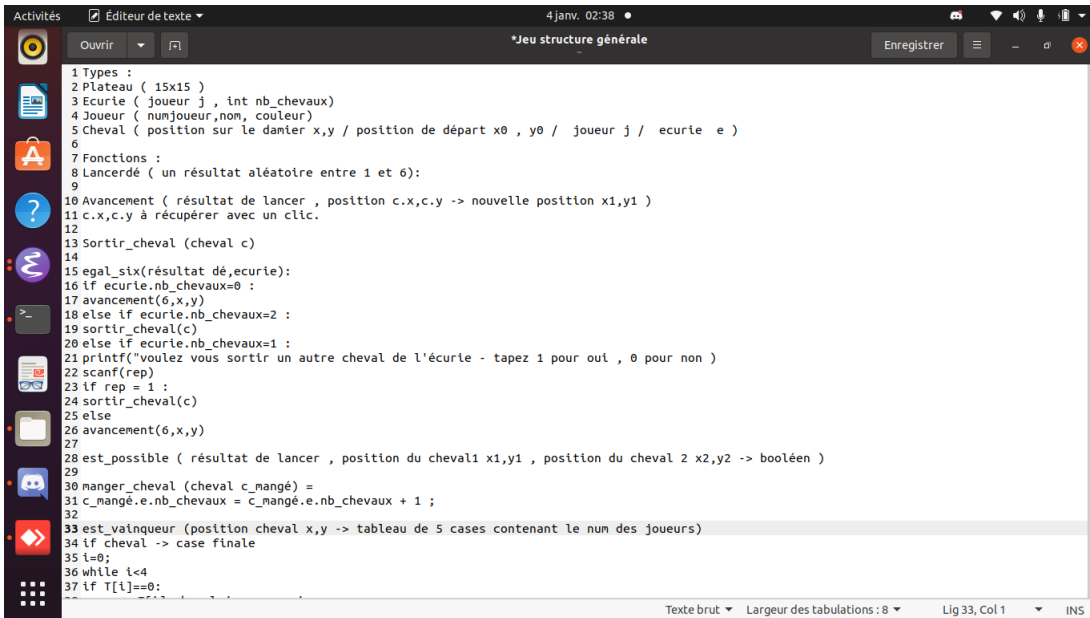
FIGURE 4 – bouton meilleur scores

NOUVELLE PARTIE

FIGURE 6 – bouton nouvelle partie

2- Premier squelette :

Avant de directement taper des lignes de code nous avons établi un squelette général de notre jeu , afin d'avoir une idée des fonctions primordiales à programmer. Le schéma général sur lequel on s'était entendu est représenté ci dessous : Bien sûr , il



```
1 Types :
2 Plateau ( 15x15 )
3 Ecurie ( joueur j , int nb_chevaux)
4 Joueur ( numjoueur,nom, couleur)
5 Cheval ( position sur le damier x,y / position de départ x0 , y0 / joueur j / ecurie e )
6
7 Fonctions :
8 Lancerdé ( un résultat aléatoire entre 1 et 6):
9
10 Avancement ( résultat de lancer , position c.x,c.y -> nouvelle position x1,y1 )
11 c.x,c.y à récupérer avec un clic.
12
13 Sortir_cheval (cheval c)
14
15 egal_six(résultat décurie):
16 if ecurie.nb_chevaux=0 :
17 avancement(6,x,y)
18 else if ecurie.nb_chevaux=2 :
19 sortir_cheval(c)
20 else if ecurie.nb_chevaux=1 :
21 printf("voulez vous sortir un autre cheval de l'écurie - tapez 1 pour oui , 0 pour non )
22 scanf(rep)
23 if rep = 1 :
24 sortir_cheval(c)
25 else
26 avancement(6,x,y)
27
28 est_possible ( résultat de lancer , position du cheval1 x1,y1 , position du cheval 2 x2,y2 -> booléen )
29
30 manger_cheval (cheval c_mangé) =
31 c_mangé.e.nb_chevaux = c_mangé.e.nb_chevaux + 1 ;
32
33 est_vainqueur (position cheval x,y -> tableau de 5 cases contenant le num des joueurs)
34 if cheval -> case finale
35 i=0;
36 while i<4
37 if T[i]==0:
```

y'aura beaucoup beaucoup de modifications pendant le codage , selon nos besoins , et aussi selon nos connaissances.

3- Typedef :

Nous avons défini différents types qui seront essentiels a notre jeu : comme par exemple

Le type cheval qui définit un pion avec sa position (un entier x et un entier y) , son avancement (le nombre de cases qu'il a parcouru à l'instant T)

Le type joueur qui possède 2 chevaux (le cj1 et le cj2 de type cheval défini juste avant) , une couleur , et une ecurie qui est initialisée à 2 vu que chaque joueurs possède 2 pions au départ).

4- Fonctions principales :

La fonction principale de notre jeu qui est d'ailleurs la première à laquelle on a commencer à réfléchir est la fonction :

- avancer_cheval :

Comme notre plateau de jeu est assez spécial , il a fallu étudier tous les cas possibles quelle que soit la position d'un pion sur les plateaux. Nous nous sommes donc intéressés aux coordonnées du pion , ça tombe bien car un pion (un cheval) est défini par un x et y qui indiquent sa position sur le plateau. Et ensuite selon sa position soit il bouge a gauche (x-) , en haut (y-), en bas (y++) , ou a droite (x++) .

- est_six :

c'est une fonction qui étudie tous les cas possibles dans le cas où le joueur obtient un 6 après le lancer de dé. exemple : - lorsque le joueur n'a encore aucun pion de sorti alors son pion 1 sort automatiquement - lorsque le joueur obtient un six et que son pion est vient tout juste de sortir (il est donc sur sa case de départ) et avance automatiquement de 6 cases car il ne peut pas faire sortir son 2ème pion. - lorsque le joueur a un pion sur le plateau (hors case de départ) et un pion dans son écurie , alors il a le choix entre faire avancer ce dernier , ou sortir l'autre - lorsque le joueur possède ses deux pions sur le plateau (donc hors écurie) on lui laisse également le soin de choisir lequel il veut faire avancer.

- retour_ecurie :

Elle remet tout simplement un cheval donné dans son écurie respective (tout dépend donc de sa couleur). cette fonction est importante car elle est appelée lorsque deux pions ennemis se croisent (on dit alors , qu'un pion en mange un autre).

- choix_user2 :

Cette fonction est appelée dans la fonction `est_six`. elle correspond au cas de figure où le joueur possède un pion hors case départ et un pion en écurie. Elle lui donne donc le choix de : soit avancer son cheval , soit en sortir un autre. le choix se fait par un clic sur un des deux boutons qui apparaitront dans ce cas de figure et qui sont : sortir cheval / avancer cheval

- Choix_user3 :

Est appelée quand les 2 pions du joueur sont sur le plateau. il a donc la possibilité de cliquer sur le pion qu'il souhaite faire avancer. on récupère les coordonnées du clic et on les compare a celles des deux chevaux , pour déterminer lequel faire avancer.

- la fonction lancer_dé :

elle retourne tout simplement un entier aléatoire entre 1 et 6 , son résultat est ensuite utilisé pour savoir de combien de cases nous allons avancer un cheval , ou si l'on doit en sortir un etc.. c'est pour cela que dans notre fonction principale nous ferons un : `l = lancer_le_de()`; et nous utiliserons l'entier `l` comme paramètre pour d'autres fonctions.

MERCI DE VOTRE ATTENTION.